



Guía Docente				
Datos Identificativos			2012/13	
Asignatura (*)	Concorrenca e Paralelismo	Código	614G01018	
Titulación				
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	2º cuatrimestre	Segundo	Obrigatoria	6
Idioma	CastelánGalegoInglés			
Prerrequisitos				
Departamento	ComputaciónElectrónica e Sistemas			
Coordinación	López Taboada, Guillermo	Correo electrónico	guillermo.lopez.taboada@udc.es	
Profesorado	Arenaz Silva, Manuel Carlos	Correo electrónico	manuel.arenaz@udc.es	
	Barreira Rodriguez, Noelia		noelia.barreira@udc.es	
	López Taboada, Guillermo		guillermo.lopez.taboada@udc.es	
	Paris Fernandez, Javier		javier.paris@udc.es	
	Quintela Carreira, Juan Jose		juan.quintela.carreira@udc.es	
	Rodríguez Álvarez, Gabriel		gabriel.rodriguez@udc.es	
Web	campusvirtual.udc.es/moodle			
Descrición xeral	Programación concorrente e paralela. Problemas típicos: interbloqueo, inanición. Semáforos. Comunicación e sincronización. Dependencias. Paso de mensaxes. Algoritmos concorrentes e paralelos.			

Competencias da titulación	
Código	Competencias da titulación

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)			Competencias da titulación
Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e a complexidade dos algoritmos propostos dende o punto de vista da concorrenca e o paralelismo.	A12		
	A20		
Coñecemento e aplicación dos principios fundamentais e técnicas básicas da programación paralela, concorrente, distribuída e de tempo real.			

Contidos	
Temas	Subtemas



T1. Principios de programación concurrente	<ul style="list-style-type: none"><li>1.1 Conceptos.<ul style="list-style-type: none"><li>1.1.1 Arquitecturas hardware</li><li>1.1.2 Sistemas operativos</li><li>1.1.3 Procesos y Threads</li></ul></li><li>1.2 Programación Multiproceso (fork/join)</li><li>1.3 Programación Multithread</li><li>1.4 Sección crítica.</li><li>1.5 Exclusión mutua</li><li>1.6 Instrucciones atómicas</li><li>1.7 Sincronización mediante condiciones</li><li>1.8 Semáforos<ul style="list-style-type: none"><li>1.8.1 Mutex</li><li>1.8.2 Semáforos</li></ul></li><li>1.9 Interbloqueo. Prevención, evitación, recuperación.</li><li>1.10 Inanición.</li><li>1.11 Comunicación y sincronización</li><li>1.12 Escalabilidad</li></ul>
T2. Algoritmos concurrentes	<ul style="list-style-type: none"><li>2.1 Productores/consumidores.</li><li>2.2 Lectores/escritores</li><li>2.3 Filósofos cenando</li><li>2.4 Compartición nula</li></ul>
T3. Principios de programación paralela	<ul style="list-style-type: none"><li>3.1 Conceptos<ul style="list-style-type: none"><li>3.1.1 Niveles de paralelismo</li><li>3.1.2 Dependencias de datos</li></ul></li><li>3.2 Modelo de paso de mensajes<ul style="list-style-type: none"><li>3.2.1 Conceptos básicos</li><li>3.2.2 Comunicación punto a punto</li><li>3.2.3 Operaciones colectivas</li></ul></li><li>3.3 Análisis de algoritmos paralelos<ul style="list-style-type: none"><li>3.3.1 Medidas de prestaciones de algoritmos paralelos</li></ul></li><li>3.4 Metodología de la programación paralela<ul style="list-style-type: none"><li>3.4.1 Descomposición en tareas</li><li>3.4.2 Asignación de tareas</li><li>3.4.3 Técnicas de optimización</li></ul></li><li>3.5 Esquemas algorítmicos paralelos<ul style="list-style-type: none"><li>3.5.1 Pool de tareas</li><li>3.5.2 Paradigma Master/slave</li><li>3.5.3 Esquemas en árbol</li><li>3.5.4 Single Process Multiple Data</li></ul></li></ul>
T4. Diseño de algoritmos y aplicaciones paralelas	<ul style="list-style-type: none"><li>4.1 Bibliotecas de paso de mensajes</li><li>4.2 Casos de estudio</li><li>4.3 Evaluación del rendimiento</li><li>4.4 Aplicación de técnicas de optimización</li></ul>

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Sesión maxistral	25	40	65
Proba mixta	4	0	4



Prácticas de laboratorio	16	24	40
Solución de problemas	10	20	30
Seminario	4	6	10
Atención personalizada	1	0	1

\*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Exposición oral complementada co uso de medios audiovisuais e a introdución de algunhas preguntas dirixidas aos estudantes, coa finalidade de transmitir coñecementos e facilitar a aprendizaxe.
Proba mixta	Exame da materia que combinará preguntas sobre a teoría con problemas a resolver.
Prácticas de laboratorio	Metodoloxía que permite que os estudantes aprendan efectivamente a través da realización de actividades de carácter práctico, tales como desenvolvemento de exercicios de programación.
Solución de problemas	Resolución de problemáticas concretas xurdidas nas prácticas de laboratorio a teor dos coñecementos que se traballaron, os cales poden plantexar múltiples solucións.
Seminario	Presentación en grupo dun tema para o seu estudo intensivo, buscándose a discusión, e a participación dos asistentes ao seminario.

Atención personalizada	
Metodoloxías	Descrición
Prácticas de laboratorio Seminario Solución de problemas	Nas prácticas de laboratorio, seminarios e na resolución de problemas os estudantes poderán presentar cuestións, dúbidas, etc. O profesor/a, atendendo a estas solicitudes, repasará conceptos, resolverá novos problemas ou utilizará calquera actividade que considere axeitada para resolver as cuestións plantexadas.

Avaliación		
Metodoloxías	Descrición	Cualificación
Prácticas de laboratorio	<p>Realización de varios exercicios prácticos de concorrencia (bloque de prácticas de concorrencia) e unha serie de exercicios prácticos de paralelismo (bloque de prácticas de paralelismo). Cada bloque de prácticas representa a metade da nota máxima de prácticas de laboratorio.</p> <p>Inda que a realización das prácticas pode ser por parellas a súa avaliación é individual.</p> <p>É condición necesaria (pero non suficiente) para aprobar a asignatura obter como mínimo o 50% da puntuación máxima de cada un dos dous bloques de prácticas de laboratorio, así como entregar e superar EN PRAZO tódolos exercicios requeridos.</p> <p>NON ENTREGAR ou NON SUPERAR en prazo a avaliación de CALQUERA das prácticas obrigatorias suporá unha cualificación de 0 no bloque correspondente de prácticas de laboratorio e a consideración de NON PRESENTADO con cualificación final na primeira edición de actas da asignatura (xuño).</p>	30
Proba mixta	<p>Proba sobre os contidos presentados na docencia expositiva e desenvolvidos nas sesións de laboratorio e nas titorías grupais. Estará dividida en dous bloques: concorrencia (temas 1 e 2) e paralelismo (temas 3 e 4), sendo cada bloque a metade da nota máxima da proba mixta.</p> <p>É condición necesaria (pero non suficiente) para aprobar obter como mínimo o 30% da puntuación máxima de cada un dos dous bloques, así como obter un mínimo do 40% da puntuación máxima da proba mixta.</p>	70



## Observacións avaliación

É condición necesaria (pero non suficiente) para aprobar obter unha cualificación global, aplicando as ponderacións da proba obxectiva e das prácticas de laboratorio, igual ou superior ao 50%. En caso de obter unha cualificación global igual ou superior ao 50% pero non cumprir con algunha das condicións necesarias a cualificación final será un&nbsp; Suspense (4.5). O mecanismo de recuperación que se habilita é exclusivamente a repetición da proba mixta (70% da cualificación global da asignatura), requeriéndose de novo como&nbsp; mínimo o 30% da puntuación máxima de cada un dos dous bloques, así como obter un mínimo do 40% da puntuación máxima da proba mixta. A cualificación de prácticas de laboratorio (30% da cualificación global da asignatura) obtida na convocatoria ordinaria consérvase para a recuperación. Non é posible recuperar/mellorar para xullo a nota de prácticas de laboratorio xa que é unha nota obtida en avaliación continua.

## Fontes de información

<b>Bibliografía básica</b>	<ul style="list-style-type: none"><li>- Francisco Almeida [et al.] (2008). Introducción a la Programación Paralela. Madrid: Paraninfo Cengage Learning</li><li>- Doug Lea (2001). Programación Concurrente en Java: Principios de Diseño y Patrones. Madrid: Addison Wesley</li><li>- Joe Armstrong (2007). Programming Erlang: Software for a Concurrent World. United States: Pragmatic Programmers</li></ul>
<b>Bibliografía complementaria</b>	<ul style="list-style-type: none"><li>- Wilkinson, B. y Allen, M.. (1999). Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers. . Upper Saddle River, New Jersey : Prentice Hall,</li></ul>

## Recomendacións

### Materias que se recomenda ter cursado previamente

Internet e sistemas distribuídos/614G01023

### Materias que se recomenda cursar simultaneamente

Sistemas Operativos/614G01016

Redes/614G01017

Proceso Software/614G01019

### Materias que continúan o temario

Programación II/614G01006

Algoritmos/614G01011

Estrutura de Computadores/614G01012

Paradigmas de Programación/614G01014

Deseño Software/614G01015

## Observacións

(\*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías